

精品推荐 ★



《软件测试求职必备宝典》

软件测试面试宝典 2022最新版

小白入门软件测试首选黑马

免费干货资料



欢迎添加黑马王伟老师企业微信
领取更多干货资料
黑马测试自助互动群也在等你哦

市场上基于不同的公司、不同的面试官，在招聘时都有各自的预期和要求。面试题可以辅助我们面试成功，但是不要作为绝对面试成功依据。这里准备了一份面试100+题（纯手码）仅供参考

一、公共题面试题

1、自我介绍

- 体现个人工作经历-几家公司、公司类型、测试项目类型（业务类型、技术类型B/S C/S /APP H5、微信小程序、公众号）
- 体现个人技能水平-独立组织策划过项目版本的完整测试过程，UI、接口自动化测试
- 体现个人对于测试的理解
- 预期或则规划

2、请简单介绍下最近这家公司你是如何开展测试工作的？

- 同类型题
 - 你们公司的测试工作流程？
 - 你是如何执行测试的？
 - 在项目中怎么测试？

1 | 这里其实就是将测试流程进行细化、丰富。这里就没有标准答案

3、描述一下你最近(简历中的项目)这个项目是做什么的？

- 体现项目是什么架构类型，包含哪些平台，分别给谁用，用来做什么？
- 体现核心的业务流程是什么？

4、描述在这个项目里面你主要负责哪些模块的测试？挑选其中一个难度大点或业务比较复杂的模块讲述你是如何测试？

- 一定是核心流程所涵盖到的模块

订单、库存（库存锁）、支付、抢购、优惠券、分销

资质、银行对接（接口阶段mock第三方接口）、利息计算、逾期计算规则、借款投资流程

进(采购)销(销售)存(库存)

- 如何测试需要体现：
 - 测试哪些业务场景
 - 采用什么方式测试
 - 对于这类型的模块一般测试几轮，测试结果如何

二、常规面试100题：

5、如何测试支付业务？

- 1 | 基于之前项目中做支付场景,当时测试的时候分成两种方式进行测试。由于我们的支付渠道:微信、支付宝、银行任意选,没有开通测试接口,只能进行真实支付。所以采用的第一种方式:在确认订单后修改订单金额为1分钱,测试支付成功后的跳转。其它异常情况正常测试即可:如余额不足、密码错误等。 基于接口测试层面,采用mock模拟接口的方式,模拟银行各种异常返回,检查我们系统对应响应处理是否正确即可。

6、给你一个网站你如何开展测试工作？

- 1 ①：首先查找需求说明、网站设计等相关文档，分析测试需求
- 2 ②：制定测试计划，确定测试范围和测试策略，一般包括以下几个部分：功能性测试，界面测试，性能测试，数据库测试，安全性测试，兼容性测试
- 3 ③：设计测试用例：
- 4 - 功能性测试可以包括，但不限于以下几个方面：链接测试；链接是否正确跳转，是否存在空页面和无效页面，是否有不正确的出错信息返回等；提交功能的测试；多媒体元素是否可以正确加载和显示；多语言支持是否能够正确显示选择的语言等
- 5 - 界面测试可以包括但不限于一下几个方面：页面是否风格统一，美观。页面布局是否合理，重点内容和热点内容是否突出。控件是否正常使用。对于必须但为安装的空间，是否提供自动下载并安装的功能。文字检查。
- 6 - （最后补充）性能测试一般从以下两个方面考虑：压力测试，负载测试，强度测试。
- 7 - 数据库测试要具体决定是否需要进行。数据库一般需要考虑连结性，对数据的存取操作，数据内容的验证等方面。
- 8 - （最后补充）安全性测试：基本的登录功能的检查；是否存在溢出错误，导致系统崩溃或者权限泄露；相关开发语言的常见安全性问题检查，例如 SQL 注入等；如果需要高级的安全性测试，确定获得专业安全公司的帮助，外包测试，或者获取支持。
- 9 - （最后补充）兼容性测试，根据需求说明的内容，确定支持的平台组合：浏览器的兼容性；操作系统的兼容性；软件平台的兼容性；数据库的兼容性。
- 10 ④：开展测试，并记录缺陷。合理的安排调整测试进度，提前获取测试所需的资源，建立管理体系（例如，需求变更、风险、配置、测试文档、缺陷报告、人力资源等内容）
- 11 ⑤：定期评审，对测试进行评估和总结，调整测试的内容。

7、没有需求文档，你会如何执行测试？

- 1 假如没有需求文档我会从以下一些方面着手：
- 2 1. 根据客户的功能点整理测试需求追溯表
- 3 2. 根据开发人员的Software Specification List整理我们的功能测试点
- 4 3. 开展项目跨部门讨论会
- 5 4. 测试人员整理用例需求疑问递交项目组和客户代表回复
- 6 5. 项目内部用例评审
- 7 6. 邮件和客户代表确认部分争议问题
- 8 7. 项目Demo和部分已开发系统
- 9 8. 参考同行业和竞争对手的类似产品
- 10 9. 交叉模块的测试要注意
- 11 10. 咨询客户部分需求疑问

8、给你一个物件（杯子、花瓶、笔、桌子）你怎么测试？

- 1 无论是哪个物件，都从以下几个维度出发设计：
- 2 1、UI
- 3 2、功能
- 4 3、性能
- 5 4、安全
- 6 5、接口
- 7 6、易用性
- 8
- 9 举例，面试官给你一个杯子你怎么测，至少写出20条测试用例：
- 10
- 11 1. 功能测试：
- 12 主要关注水杯基本功能
- 13 1.1 水杯是否可以正常装水
- 14 1.2 水杯是否可以正常喝水
- 15 1.3 水杯是否有盖子，盖子是否可以正常盖住
- 16 1.4 水杯是否有保温功能，保温功能是否正常保温

17 1.5 水杯是否会漏水，盖住盖子拧紧后是否会漏水
18 2.ui测试：
19 主要关注水杯外观、颜色、设计等方面
20 2.1 外观是否完整
21 2.2 外观是否舒适
22 2.3 颜色搭配及使用是否让人感到舒适
23 2.2 杯子外观大小是否适中
24 2.3 杯子是否有图案，图案是否易磨损
25 3.易用性测试：
26 主要关注水杯使用是否方便
27 3.1 水杯喝水时是否方便
28 3.2 水杯拿起放下是否方便，这里会衍生到水杯形状的测试
29 3.3 水杯装水是否方便
30 3.4 水杯携带是否方便
31 3.5 水杯是否有防滑功能
32 3.6 水杯装有低温或者高温水时，是否会让手感到不适
33 4.性能测试：
34 4.1 水杯装满水时，是否会漏出来
35 4.2 水杯最大使用次数
36 4.3 水杯的保温性是否达到要求
37 4.4 水杯的耐寒性是否达到要求
38 4.5 水杯的耐热性是否达到要求
39 4.6 水杯掉落时，是否可以正常使用
40 4.7 水杯长时间放置时，是否会发生泄露
41 5.安全性测试：
42 主要关注水杯外观和各种异常条件下是否释放有毒物质等
43 5.1 当水杯装满热水时，水杯是否会烫手
44 5.2 当水杯装上水后，是否会产生有毒物质
45 5.3 把水杯放在零下环境时，是否会产生有毒物质
46 5.4 把水杯放在高温环境时，是否会产生有毒物质
47 6.接口（杯子没有想到怎么和接口关联起来）
48 7.兼容性测试：
49 主要关注水杯是否可以装其他液体，如果汁、汽油、酒精等
50 8.可移植性测试：
51 主要关注水杯放置环境等
52 6.1 将水杯放在常温环境中，使用是否正常
53 6.2 将水杯放在零下的环境中，使用是否正常
54 6.3 将水杯放在高于正常温度的环境中，使用是否正常
55
56 你看，这道面试题是不是就轻松解决了？

9、如何执行的兼容性测试？-web兼容和app兼容

1 (1) web 兼容性测试

①首先开展人工测试，测试工程师测试主流浏览器和常用操作系统测试主流程和主界面，看看主流程和主界面是否有问题，如果存在问题，那么记录下 bug 情况，以及浏览器型号和版本，以及操作系统，准确定位 bug 产生的原因，提交 bug，告知开发人员修改。所有的主流设备都需要进行测试，只关注主流程和主界面，毕竟每个系统主流程和主界面不是很多，所以这个工作量还是可以承受的。

2 (2) APP 兼容性测试

①：APP 的兼容性测试和 web 测试类似，首先开展人工测试，测试工程师借助测试设备对主流程和主功能，主界面进行测试；收集所有的能收集到的不同型号的测试设备测试主流程和主界面，看看主流程和主界面是否有问题，如果存在问题，综合考虑设备的使用率等因素，看看是否需要调整，如果需要，那么记录下 bug 情况以及测试设备的型号和操作系统，准确定位 bug 产生的原因，提交 bug，告知开发人员修改。

②其次借助第三方测试工具，对于 APP 的兼容性测试，推荐的是百度众测平台和云测平台，我经常使用的是云测平台，这两款测试工具里面包含了安卓和 ios 的测试；测试很齐全，包括功能测试、深度兼容测试、性能测试、网络环境测试，还可以模拟海量用户测试，还可以导入自己编写的测试用例进行功能测试，里面还包括测试专家的测试，当然了找专家是要花钱滴。基本进行兼容性测试是不需要花钱的；测试工程师把打包好的 apk 或者 IPA 文件，上传到测试平台，选择需要测试的设备型号，开始任务即可；等待一段时间，在等待的时间你是不需要盯着的，你可以做其他的工作。测试完成后会生成一份测试报告，可以查看错误页面和错误日志，如果需要调整，那么提交 bug，告知程序员修改即可。

10、你们团队人员比例如何？

1 一般情况下比例不要小于 开发:测试= 5:1 团队人数最好不要超过30

11、需求发生变化，你会如何处理？

1 分两种情况

2 1、本次需求变更的内容对应模块还没有进行测试

3 1.1 评估需求变更对应模块所带来的工时是否大于原来该模块的测试工时

4 1.2 如果超过原有模块的测试工时，影响测试计划则需要考虑以下几点，看是否的能在原计划的范围内内部消化：加班、协调资源、根据现有测试过的模块情况来分析是否可以调整一些测试时间、如果所有方式都已经考虑则反馈给测试负责人或项目经理来进行协商

5 2、本次需求变更的内容对应模块已经完成了测试

12、多久做一次版本迭代？如何进行回归测试？

1 一般分大版本和小版本，大版本主要是产品规划的新功能、新业务，小版本主要是一些历史功能优化和缺陷修复版本。大版本一般2-3个月一次。小版本每周都会有。

2 1、缺陷回归：触发缺陷查看缺陷是否已经修复；

3 2、历史功能回归：跟项目经理以及开发确认本次版本迭代影响的功能范围，对于影响的功能范围以及核心业务流程、关键点，挑选正向的用例进行回归测试；同时利用版本迭代的空闲时间，对历史功能回归测试实现UI自动化

13、在测试过程中有没有发现过让你记忆犹新的BUG？

1 这个问题不一定难，重点体现真实

2 问题描述:购物车合计金额和订单的商品合计金额偶尔出现相差0.01

4
5 需求描述:基于商品合计金额结果四舍五入
6 购物车合计金额 = $\text{sum}(\text{购买量} * \text{单价} * \text{会员折扣率})$
7 订单的合计金额 = $\text{sum}(\text{购买量} * \text{单价} * \text{会员折扣率}) + \text{物流费} - \text{优惠券} - \text{积分抵扣}$
8
9 分析问题:
10 找到对应购物车研发人员确认其计算的方式,每次计算完一个乘法四舍五入
11 订单的开发人员计算方式是乘法计算完之后再对于最终结果进行四舍五入,导致偶尔出现相差0.01的情况
12
13 解决方式:找产品确认最终确认进位方式
14
15 这里就提供一个,大家在实际学习过程中也会遇到一些缺陷。可以进行记录和分析。

14、提交了一个缺陷开发不认为是缺陷你会如何处理？

1 个人在以前的测试过程中,很少出现这种问题。一般情况下我提交缺陷前都会反复的确认,也会尽量的保障缺陷有效率
和清晰度。如出现该情况,首先我会找到对应的开发人员,询问对方拒绝或不认为是缺陷的理由。如在对方的描述过程
中发现明显的问题,且有明确的证据情况会摆明需求甚至当场重新给其确认。如在其阐述理解的过程中,没有问题,或者
说其在对于需求的理解中也没有明确的错误,那证明对于需求的理解出现的了歧义,直接找产品或项目经理确认达成共
识即可。当然也可能是其它原因导致缺陷的无效,例如垃圾数据、环境问题等,这些情况一般出现很少,一般在我测试
过程中对应的测试数据我都会保障其有效性,环境问题也不多,就是有时候测试的版本可能不是最新的代码,一般情况下
我们的测试团队每天测试开始前,都会去更新最新的代码来进行测试,但是不能避免有最新更新,这块实时多同步下即
可。

15、测试时间不够了,项目又必须上线,你会如何处理？

1 这里测试时间不够了,可能导致原因有很多,如开发或缺陷修复质量不高,导致无法顺利执行测试。第二需求的变更导致
计划控制不合理等等,但是不管那种原因,客观时间不够了是事实。那我的话一般会做如下方式来进行处理:第一,迅速
盘点剩余测试工作量,是否可以通过加班或协调资源的解决问题,保障项目上线。第二,是否可以根据项目各个模块的现
有测试情况来调整原有测试计划的分配工时,如部分模块无较大问题,是否在风险可控的情况减少测试工时,第三,是否
可以借调产品和开发分摊部分简单的测试任务,第四,原有复杂的业务功能是否有替代上线方案,第五,是否可以占用部
分产品验收时间,第六,是否可以裁剪部分功能

16、如何保障测试用例的覆盖率？

1 1.编写测试用例前,先熟悉项目,看看相关需求文档是否有问题(功能描述不清,设计逻辑缺陷),如有问题找相关设
计或者开发问清楚。
2 2.采用测试用例设计方法:如判定表、等价类、边界值、流程图等
3 3.组织用例评审,进行查漏补缺

17、生产环境中有没有出现比较严重的问题?如果有的话有什么的方法可以降低带来的风险？

1 1.在之前的项目测试经验中,线上暂时没有出现过比较严重的问题。
2 2.如果出现严重的缺陷的话,我会思考一下的一些问题,看是否能降低风险:
3 a.第一:先确认导致问题出现的原因,如果是代码逻辑错误,证明漏测-则需要加强测试力度、加强测试用例评审力
度
4 b.第二:如果是遗漏代码,测试后提交,加强代码封板把控,而上线前必须同步最新代码,测试确认。
5 c.第三:如是需求实现错误,加强需求评审力度。
6 d.第四:采用灰度上线,大型版本先只对部分用户开放,稳定后再全面升级。

18、你们有多少个测试环境,分别有哪些？

- 1 之前的项目中有三个测试环境三个，测试环境，回归环境，预发布环境
- 2 测试环境，也就是我们测试同学干活的环境啦，一般会由测试同学自己来部署，然后在此环境进行测试。bug修复后，需要发版更新测试环境来回归bug。
- 3 回归环境，回归bug的环境，其实就是我们的测试环境，在测试环境上测试、回归验证bug。
- 4 预发布环境，测试环境到生产环境的过渡。测试环境可能会受到一些限制，一些流程或者数据没有测试到，就可以在预发布环境进行验证，从而保证产品上线质量软件测试环境包括硬件环境和软件环境，

19、有没有部署过测试环境，简单的描述下如何部署的？

- 1 有！
- 2 在之前的项目中，我们的系统使用的应用服务架构是：Linux+Nginx+Tomcat+Mysql。
- 3 在进入测试前，开发已经将整套的服务器架构都已经搭建好，所以对应的中间件和工具环境不需要再重新部署。当时我去部署的过程，主要是更新一些新的测试版本。会从git上面获取到开发的源代码到我的本地电脑，并使用开发提供的编译打包命令，进入到源码所在路径，执行mvn clean install即可获取得到一个可部署的源码包，(TARGET)*****（可以自己起名，例如叫自己的项目的英文缩写）。war文件。通过SSH远程连接工具MobaXterm将源代码上传到Linux测试服务器。停止测试环境tomcat服务，备份原测试war包。将刚上传的***.war移动至tomcat指定的项目部署目录webapps，重启通过tomcat/bin目录下的startup.sh启动服务器。最后到本地机器访问测试域名地址，如能成功访问则代表部署成功了。
- 4
- 5 另外我还了解过jenkins可以执行持续构建部署方式。如有需要可以进行简单运用。

20、了解哪些网络协议，简单描述一下区别？

- 1 在之前的测试工作中主要是测试https和http协议，当然简单了解过TCP协议：
- 2 1、https协议需要到CA申请证书，一般免费证书较少，因而需要一定费用。
- 3 2、http是超文本传输协议，信息是明文传输，https则是具有安全性的ssl/tls加密传输协议。
- 4 3、http和https使用的是完全不同的连接方式，用的端口也不一样，前者是80，后者是443。
- 5 4、http的连接很简单，是无状态的；HTTPS协议是由SSL/TLS+HTTP协议构建的可进行加密传输、身份认证的网络协议，比http协议安全。

21、常见的响应状态码有哪些？分别表示什么含义？

- 1 2开头：2xx（成功）表示成功处理了请求的状态代码；如：200（成功）服务器已成功处理了请求。
- 2 3开头：3xx（重定向）表示要完成请求，需要进一步操作。通常，这些状态代码用来重定向。如：304（未修改）自从上次请求后，请求的网页未修改过。服务器返回此响应时，不会返回网页内容
- 3 4开头：4xx（请求错误）这些状态代码表示请求可能出错，妨碍了服务器的处理；如：400（错误请求）服务器不理解请求的语法；403（禁止）服务器拒绝请求。404（未找到）服务器找不到请求的网页。
- 4 5开头：5xx（服务器错误）这些状态代码表示服务器在尝试处理请求时发生内部错误。这些错误可能是服务器本身的错误，而不是请求出错；如：500（服务器内部错误）服务器遇到错误，无法完成请求。

22、系统出现500或白屏，你会如何分析问题？

- 1 查看系统服务器资源是否占满：磁盘、内存
- 2 2、通过查看磁盘占用状态：df -h；通过写入缓存信息，导致硬盘空间不足；
- 3 3、查看内存的使用情况：top
- 4
- 5 - 内存泄漏：由于开发编写代码过程对于已经分配内存资源使用完毕之后，并不释放内存的资源，导致后面同样业务处理所占内存足部累加，最终导致内存不足。
- 6 - 内存溢出：现有空闲内存不足以提供服务器处理客户请求。
- 7
- 8 作为测试需要定位信息：
- 9 基于内存泄漏，需要找到占用了资源而不释放具体使用功能是哪个，然后基于功能找到请求地址
- 10

23、Fiddler主要用来做什么？如何分析抓取的数据信息是正确的？如果是加密的内容你们怎么处理？

- 1 1、分析缺陷是前端的问题还是后端的问题。
- 2 例如：提交订单的请求地址：
- 3 ① 界面组织提交订单商品数据，点击【提交订单】，触发发送请求
- 4 ② 后台代码进行处理，处理完成之后，返回订单相关数据。返回的数据由开发者来决定（需求来决定到底返回哪些数据）订单编号、订单金额
- 5 例如订单金额在界面显示错误，抓取提交订单响应数据，查看接口返回信息中订单的总额是否正确。如果接口中订单总额正确，则是前端的问题，如果是响应信息中订单总额是错误，则是后端的问题。
- 6 2、前端对于输入信息做了对应限制，不代表后端代码也做了限制，每个请求地址对应懂IT的人的来讲都是能够直接跳过前端页面进行操作的。验证后端对于异常输入的是否也有做对应限制。
- 7 3、接口测试测试每个请求的实现情况。部分公司的开发没有编写接口文档，则可以通过抓包工具获取到具体接口地址。
- 8 4、做手机端弱网测试

24、有没有做过弱网测试，简单的说下？

- 1 我会让PC端和移动端连上相同的网络，Fiddler通过代理连上手机，PC端通过Fiddler工具的选项卡Connerctions设置Fiddler监听端口为8888，移动端修改网络设置将IP地址和端口设置为电脑的IP地址和Fiddler的中的端口号，安装Fiddler证书。移动端就可以打开要进行弱网测试的网页或者APP，Fiddler会自动获取相关的地址，通过修改customRuler.js文件里的m_SimulateModem的参数来模拟弱网，从而就可以执行弱网测试了。

25、Fiddler如何抓取苹果手机包？描述下过程？

- 1 基于苹果手机抓包：
- 2 1、在Fiddler配置项中要设置勾选抓取HTTPS，自动下载HTTPS证书
- 3 2、开启网络服务监听端口：勾选Allow remote computers to connect，端口号可默认为8888
- 4 3、打开iphone浏览器，在地址栏输入http://ip(电脑的ip):8888(fiddler中默认的端口)，点击“FiddlerRoot certificate”安装证书
- 5 4、设置iphone代理：手机连接wifi后，打开wifi详情，滑动至屏幕底部的http代理，点击进入修改成“手动”，设置ip（电脑的ip）及端口（fiddler中的8888）后保存启用证书方法：设置—通用—关于本机—证书信任设置

26、用过哪些类型的数据库，关系型数据库和非关系型数据库有什么区别？

- 1 用过MySQL、Oracle了解redis数据库
- 2 关系型数据天然就是表格式的，因此存储在数据表的行和列中，数据表可以彼此关联协作存储，也很容易提取数据。与其相反，非关系性数据不适合存储在数据表的行和列中，而是大块组合在一起，非关系型数据通常存储在数据集中，就像文档，键值对或者图结构，你的数据集体特性是选择数据存储和提取方式的首要影响因素。

27、如何使用SQL快速插入100000条数据？


```

1 DELIMITER // # 使用DELIMITER关键字临时声明修改SQL语句的结束符为//
2 create procedure test() # 创建存储过程
3 begin
4 declare i int default 0; # 声明一个默认值为0的局部变量i
5 while i<100000 do # 开始循环
6 insert into books(name)value("test"); # books是表名, name是字段名 test是字段值
7 set i=i+1; # 使用set为参数赋值
8 end while;
9 end //
10 DELIMITER ; # 将结束符重新定义回结束符为";"
11 call test(); # 调用函数

```

28、写过最复杂的SQL语句，表名叫什么？

1 这个题目不同项目自行准备，3表以上关联、分组、排序查询

29、数据库还可以用来做什么？

1 1 执行测试用例时，有时需要到数据库验证数据的准确性与完整性
2 2 进行bug定位时，有时需要到数据库查看数据的详细信息
3 3 构造某种测试场景时，可以在数据库里直接修改数据，要比使用界面更有效率
4 4 软件升级过程中，经常会涉及到对历史数据的处理，这种情况需要执行升级sql，并验证结果。

30、为啥选择做软件测试？不选择IT其它岗位，如开发？

1 这个题自行发挥，能体现出对测试行业的热爱即可

31、有没有制定过测试计划？测试计划包含一些什么内容？依据什么来制定的测试计划？

1 在我们的项目中测试计划主要涵盖：测试范围、测试资源、任务分配、阶段任务完成时间节点，偶尔会加上风险评估、测试环境等相关说明。
2
3 一般情况下会依据需求文档确认测试范围的测试点拆解文案、开发计划文档、产品/运营的目标上线时间以及当前测试人力资源情况来制定测试计划。

32、你们公司是如何管理的缺陷？缺陷处理的过程是怎么样的？

1 1. 测试人员发现并确认缺陷，在禅道记录缺陷，将其指派给指定模块开发人员。如优先级或缺陷等级比较高，同时会直接通知对应开发人员尽早修复。
2
3 2. 开发人员到禅道领取缺陷，分析缺陷并进行处理。当开发人员进行处理并认为已经解决之后，就可以将这个缺陷的状态设置为：已修复，并将其返还给测试人员。
4
5 3. 测试人员进入系统查看缺陷，并测试验证缺陷。如果经过再次测试发现缺陷仍然存在的话，测试人员将缺陷再次传递给开发人员，并将缺陷的状态设置为“重新打开”。如果测试人员经过再次测试确认缺陷已经被解决，就将缺陷的状态设置为“已关闭”。
6
7 4. 如果测试经理收到某缺陷被拒绝通知，验证该缺陷，如果确实不能算作缺陷，关闭缺陷，将缺陷状态设置为“已关闭”。如果认为的确是一个缺陷，修改缺陷描述，并将其重新指派给开发经理，并将缺陷的状态设置为“新建”。

33、如何评估缺陷的优先级和严重级别？

- 1 <不需要全部都回答，大概描述即可>
- 2 优先级：指开发修复缺陷的先后顺序，严重级别：指发现的缺陷对产品质量影响的严重程度
- 3
- 4 高（P1）：缺陷严重，影响测试或产品目的，需优先考虑。
- 5 如：产品的版权未更正、技术性的不正确内容等。应用程序包版本不对、无法安装导致无法测试、闪退、崩溃等
- 6 中（P2）：对产品来说不是那么关键的场景或特性。
- 7 如：在小标题上发现的错误、从历史版本中来的遗留bug等。
- 8 低（P3）：对产品使用没有太大影响。
- 9 如：在低使用频率页面中发现的bug、很少被使用的功能等。文字重叠、错别字等UI方面的缺陷，不易操作等。
- 10
- 11 缺陷的严重级别
- 12 A类-致命缺陷（Fatal）：造成系统或应用程序崩溃、死机、或造成数据丢失，主要功能完全丧失，导致本模块以及相关模块异常等问题。
- 13 B类-严重缺陷（critical）：系统的主要功能部分丧失、数据不能保存。如致命的错误声明，程序接口错误等约束条件
- 14 C类-一般缺陷（major）：次要功能没有完全实现但不影响使用。如提示信息不太准确，或用户界面差，操作时间等，
- 15 D类-轻微缺陷（Minor）：使操作者不方便，但它不影响功能过的操作和执行，如错别字、界面不规范
- 16 E类-意见优化（Enhancemental）：由问题提出人对测试对象的改进意见。
- 17 要注意的是，开工会根据优先级来进行参考，决定修复先后顺序。所以高优先级不一定是致命的缺陷

34、你们公司测试通过的标准是什么？

- 1 1、需求规格说明书中的需求必须全部实现并测试通过
- 2 2、测试用例执行率100%
- 3 3、无一般、严重、致命等级缺陷
- 4 4、遗留缺陷和产品或项目经理沟通可以延迟处理

35、生产环境发现缺陷你会如何处理？

- 1 (1) 在公司的缺陷管理系统（如：禅道）中记录该缺陷
- 2 1. bug等级
- 3 2. 提供必要的截图和日志log信息(tail -200f log.txt)
- 4 (2) bug信息定位（前端bug还是后端bug，sql问题？）—— 缩小问题的范围
- 5 (3) 在测试环境去复现bug
- 6 1. 不能复现：可能是版本升级导致的，（联系运维、产品）是否需要需要回退到上一个安全稳定的版本
- 7 2. 能复现：召集开发、产品、运维预估 bug 解决时间
- 8 3. 解决时间短：尽快修复，尽快回归，尽快线上验证
- 9 4. 解决时间长：联系运维回退到上一个安全稳定的版本
- 10 (4) bug 分析总结
- 11 1. 测试不到位（更新测试用例）
- 12 2. 紧急变更（争取测试资源包含测试时间）
- 13 3. 环境软硬件差异性（增加线上回归测试用例）

36、在你的项目（基于自己的项目）中编写了多少条测试用例？发现了多少缺陷？

- 1 1年左右的的项目：在上一个项目里我一共是编写1300多条测试用例，其中一共是发现了300条左右的bug

37、你们团队是如何同步工作情况？或如何协同作业的？

- 1 - 需求变更时如何同步
- 2 - 提价产品验收, 协助产品测试, 遗留缺陷到验收解决
- 3 - 和开发对接-开发模块完成的时间(提测时间, 提测的方式和标准)
- 4 - 提交缺陷如何通知开发, 如何处理

38、你如何评估一个项目或版本的质量好坏? 项目上线的标准是什么?

- 1 从功能性、易用性、可靠性、兼容性、可维护性、可移植性这几个维度来判定软件质量的好坏
- 2
- 3 上线标准
- 4 (1) 软件需求分析说明书中定义的所有功能已全部实现, 性能指标全部达到要求。
- 5 (2) 在验收测试中发现的错误已经得到修改, 各级缺陷修复率达到标准
- 6 (3) 所有测试项没有残余紧急、严重级别错误。
- 7 (4) 需求分析文档、设计文档和编码实现一致。
- 8 (5) 验收测试工件齐全(测试计划、测试用例、测试日志、测试通知单、测试分析报告, 待验收的软件安装程序。)

39、在你们公司是如何执行的需求评审工作? 有没有提出过问题? 需求评审过程中测试人员应该注意什么?

- 1 - 会议前, 先明确参与会议的时间、地点和人员, 并进行通知, 开发和测试需提前熟悉需求, 标记不清楚, 疑问, 建议的地方, 等待会议时一并提出, 提高会议效率
- 2 - 会议中, 提出对需求的疑问和建议, 并对提出的问题做记录, 以免遗漏, 按需采纳修改建议, 保证参与人员对需求理解一致。当存在的异议点较多时, 考虑是当下解决还是例外安排会议进行解决。
- 3 - 会议后, 产品按评审意见对需求文档进行修改, 最后形成统一的、标准的需求文档发出来确认。若评审不通过, 那么可能就要修改后进行二次评审, 保证需求的完整性、准确性、理解一致性。
- 4 - 要保证会议的高效进行, 注意控制会议时间、讨论重点等, 以免进度拖沓, 重点问题得不到有效解决
- 5

40、使用ADB做过什么? 举例说明下? 如何获取和分析手机日志信息?

- 1 1、运行设备的shell(命令行);
- 2 2、管理模拟器或设备的端口映射;
- 3 3、计算机和设备之间上传/下载文件;
- 4 4、将本地apk软件安装至模拟器或android设备。
- 5
- 6 `adb logcat > fileName.txt`
- 7 此命令可以直接捕获全局的日志信息, 开发人员可以通过分析日志获得自己想要的信息
- 8 `adb logcat | grep <过滤内容> > fileName.txt`
- 9 此命令可以过滤内容

41、使用monkey测试了什么内容? 测试过程中有没有发现问题?

- 1 使用monkey验证app在大量随机性操作的下时候是否会崩溃, 主要是用于验收测试阶段。
- 2
- 3 测试过程中是有发现问题的, 在我之前的项目中, 在使用monke执行第二次压力测试时, app突然崩溃, 当我查看日志信息显示NullPointerException, 并记下seed值, 将错误复现了一遍, 发现依然报错, 就截图发给对应的开发确认是什么问题导致的
- 4

42、使用ABD如何连接手机? 冷热启动如何测试的? 你们的app启动时间多长?

```
1 用Adb有两种连接手机的方式，在测试过程中，各有优缺点吧。
2      一种是用USB连接
3          优点：插上就可以连接，延迟低。
4          缺点：需求借助USB线来连接，做兼容的自动化测试的时候比较麻烦
5              1.会在手机上打开开发模式，启用USB调试。
6              2.USB线连接手机和电脑
7              3.在cmd窗口输入 adb devices (手机上会有一个提示：是否允许此电脑进行调试，确认)
8      另一种是用WIFI连接
9          也有优点：使用无线连接，做自动化的时候可以远程运行
10         缺点的话：需求借助网络，延迟高，速度略慢（并没有感觉到）
11             1.先用USB连接手机和电脑
12             2.在cmd窗口输入 adb tcpip 5555
13             3.拔掉usb线
14             4.查看手机IP地址 或者直接在CMD窗口输入命令
15                 adb connect IP:5555
16
17         冷启动：当进程不存在的时候，从进程创建开始到界面的展示的过程；
18         热启动：大部分资源都在，只是应用之间的切换；
19         adb shell am start -w 包名/界面名
20
21         冷启动：需要1秒甚至更长；
22         热启动：需要<0.5秒；
```

43、你常用的Linux命令有哪些？分别说明下用途和命令？5个以上

```
1  ps -aux | grep bin: 查看系统进程,并查找特定应用的程序
2  -top: 用于实时查看系统进程相关信息
3  -kill -9 进程号: 强制关闭对应的进程
4  netstat -anptu |grep mysql: 查看特定进程的端口信息
5  tail -f 文件名: 实时显示日志的文件信息
6  tar -zxvf 打包文件.tar.gz -C 目标路径: 解压压缩文件
7
8  vim编辑器一定要知道,以及vim中的一些常用快捷键
```

44、在测试环境出现偶尔出现的BUG，你会如何处理？

```
1  - 当bug出现时,抓取log或截图,视频留下证据
2  - 分析问题严重级别,以及同步开发进行分析修复时长,测试评估测试时长。如果影响范围大且修复时间较长,则建议回滚代码。否则直接修复验证并重新发布
3  - 记录前置环境,操作步骤,和使用过的数据, 尝试重现bug
4  - 若无法重现, 则找开发人员协助重现; 在开发人员的协助下仍未能重现, 则考虑bug的实际影响定义bug的严重级别, 是否能延迟到以后的版本再来修复
```

43、在你的项目中做过UI自动化，简单说说你是如何做的？

- 1 该问题根据自己的实际情况还选择实施的程度:1、只会Selenium基本API编写面向过程脚本;2、能使用单元测试框架组织测试用例,不需要任何封装;3、能编写完整的自动化测试框架
- 2 有!
- 3 程度1:有简单的了解过web自动化脚本的编写,使用的是Selenium工具,在之前的项目里面基于***功能编写了一些自动化面向过程的脚本,利用元素定位和常用API实现模拟手工操作。利用这些脚本可以在测试过程中去代替一些手工构造测试数据的过程。对于单元测试框架unittest/Pytest有一定得了解,可以参考一些文案写出简单的单元测试用例。
- 4
- 5 程度2:在***项目中,基于自己所负责的模块有利用自己的时间,使用单元测试框架Pytest+Selenium编写过一些功能的自动化测试脚本,例如实现主流程的自动化测试用例,***功能的测试用例。并使用单元测试框架中Parameter实现了参数化,以及assert断言。可以利用Pytest实现批量执行测试用例,结合Allure擦肩生成测试报告。
- 6
- 7 程度3:在***项目中,由于经常有版本迭代,回归测试任务还是比较多的。部分核心的功能模块界面也比较稳定,为了提供回归测试效率,团队采用了UI自动化测试技术来解决回归测试的问题。我们使用的实现技术框架为:pytest单元测试框架+Selenium/Appium+PO页面对象的封装模式+Allure测试报告的框架来实现的UI自动化。整个项目过程中,也没有专门规划一段时间去一次性完成整体的自动化,而是测试经理规划好整体实现框架后,在版本迭代空闲时间,分配一些脚本编写的任务。在我离开公司时,UI自动化测试用例库已经基本覆盖到了项目核心功能点和流程。如***流程、***功能模块。其中***是我编写的。 ---->面试官追问细节PO,请跳转PO模式面试题

44、最常用的元素定位方法是什么?如果元素定位不到你会如何分析?

- 1 最常用的元素定位方法是XPATH,当然也会根据实际情况用一些其它的定位方式,例如:CSS等
- 2
- 3 定位不到元素无非以下几种情况,一般情况下我会从简单的开始分析:1、先确认是否定位信息写错,2、再确认是否有窗口切换;3、查看元素是否在iframe标签中;4、元素等待的问题;5、定位代码前面的步骤就有问题,导致没有到所要定位元素界面等等。基本上定位不到元素的问题都可以解决掉。

45、显示等待和隐式等待的区别是什么?哪个用的多?为什么?

- 1 显示等待和隐式等待的核心区别在于:隐式是一个全局设置,显示只针对单个元素;隐式等待受整个页面加载的影响,而显示等待只要所要定位的元素加载到就可以继续运行。还有实现方法、超时异常有区别。
- 2
- 3 其实都会用,隐式等待本身就只有设置一次,在创建浏览器驱动对象的时候就会加上。而显示等待会基于对Selenium元素定位的方法进行二次封装,后续调用元素定位方法时都会调用二次封装后的元素定位代码。所有的元素定位都会对于显示等待生效。而有隐式等待多了一重元素等待的保障。

46、有没有做过二次封装?封装了哪些方法?简单的描述下?

- 1 有的!例如元素定位方法,会进行显示等待,会兼容8种元素定位的方法的封装;如模拟输入:基于原有的send_keys方法输入之前,加上模拟清除的操作;再例如窗口切换、select下拉框等多行实现的操作方法都会进行二次封装。这些封装的代码我们都会放在一个基类中统一的进行管理。

47、桌面应用如何实现的自动化?有没有解决方案?

- 1 由于在课上没有教这块的相关实现知识,所以学员需要根据自身的情况来说。可以不会,有了解也是一种优势。
- 2
- 3 回答:我之前所实现的项目是web端的项目和APP,暂时还没有测试过桌面应用的相关软件程序。所以基于桌面应用的UI自动化实现方式没有具体实践过。但是我知道市场有可以实现该UI自动化测试解决方案和工具:如QTP就可以实现桌面应用,最新AIRTEST也可以。个人感觉其实现的大体思路和web端的大同小异,只是页面元素换成了桌面元素,如果公司有需要,我可以利用自己的时间去研究。

48、UI自动化解决了什么问题?你觉得其价值大不大?主要体现在哪里?

- 1 个人觉得UI自动化测试核心解决的问题还是解决手工回归测试,基于历史功能或流程进行回归,代替绝大部分的回归测试工具,缩减测试工程的重复的工作。第二:UI自动化的脚本还可以通过运行来准备测试数据,例如我要测试生成订单后的各种流程,就可以通过运行生成订单的测试用例。
- 2
- 3 我觉得UI自动化测试脚本如果编写的比较全面的话,价值还是很大的。就例如上面说的回归测试和构造数据,不过前提是项目适合做UI自动。否则就是白白投入。

49、如何保障测试用例可以无限运行?如何解决测试数据的问题?

- 1 我觉得这个问题核心是想说数据的重复使用吧。例如新增商品我必须保障测试用例每次都是新增的不同商品名称,否则第二次新增就会出现业务限制商品名称不能重复。再例如购物车提交订单,必须保障购物车有商品数据。我会使用两种方式:类似如数据不能重复的问题,对于录入的信息可以使用时间戳来随机生成数据保障每次都不一样。需要前置数据的:可以通过每次运行前重置测试数据库的方式。也可以通过调用其它加入购物车的形式来保障前置数据。这样脚本就可以无限运行了。

50、微信小程序如何执行UI自动化测试?

- 1 首先基于小程序而言,个人不觉得没有必要的条件的話没有必要做UI自动化。如有需要可以使用APPIUM+PYTEST的形式来进行实现。微信小程序一般是基于微信客户端的层面上进行研发的。还需要获取小程序的实现形式,一般非原生态APP。在实现的时候还需要获取webview相应信息,现在基于浏览器获取的形式还需要翻墙才能实现。然后通过上下文切换后,后续元素定位和操作的方式和常规的UI自动化实现形式没什么却别。

51、简单的说下PO模式的设计思想?

- 1 所谓的PO吧,就是Page Object页面对象,利用面向对象的思维来封装我们的UI自动化测试用例。使用这个框架的原因是:原元素定位的代码写在单元测试框架中,如登录的场景每次都要写几行元素定位和操作的代码,我们所要测试的场景很多都要登录,那每个测试用例都需要写这几行,如某天前端开发修改了其中的一个定位信息,那所有用到该定位的代码都需要进行修改。不方便维护了。而PO模式就是把我们的测试过程中每个页面都当做对象,在我们的UI自动化测试框架中,每个页面都有一个独立的封装py文件,分为三层来管理:第一层是对象库层,将测试用例在该页面所有用到的元素对象统一找到并进行管理,第二层业务层:基于每个元素定义好其对应的操作方法,第三层:业务层,组装多个操作方法即可形成完整的业务操作。例如:登录页面:对象库层找到用户名、密码等元素,操作层封装用户名输入、密码输入等方法,业务层连续调用操作层方法,那调用业务层的登录方法传递输入参数即可完成登录。这样无论哪个测试用例需要进行登录都只需要调用该页面py文件业务层的登录方法即可完成登录了。这样整个UI自动化测试代码中,对应的元素定位代码只会存在一份了,方便了维护,也减少了代码的冗余。

52、你的项目中用的什么单元测试工具?unittest和pytest的区别是什么?

- 1 unittest和pytest任选其一相对而言:unittest和pytest都是做单元测试的工具,其核心的思路是主体结构都是一样的。非说区别的话,pytest更偏向于命令行的运行形式,有独立的配置文件所以在结合持续集成时更为方便。第二pytest有更加丰富的插件,如失败重试、排序等;第三:pytest在定义测试类时不需要继承。

53、简单的描述下数据驱动实现的方式?数据存储在哪里?

- 1 我们之前项目数据驱动实现的方式是利用json和parameterized参数化插件实现的。json文件用来组织定义测试数据,parameterized用来遍历数据,读取json数据通过定义一个函数来进行读取。当然我还了解数据的存储方式还有其它很多种形式:可以直接去读取数据库的数据、也可以将测试数据定义到excel文件中。具体看需要

54、APP测试和WEB端测试的区别有哪些?

- 1 web端和app端如果是同个产品的不同平台的话，基于同样的功能模块，业务逻辑方法和ui界面的测试方法是没有任何区别的。而不同的位置是APP是C/S架构，所以会进行安装/卸载和升级测试，还有APP的运行网络环境不一样，会测试各种网络情况，甚至进行弱网测试，APP兼容性、交叉事件测试等不同。APP本身对手机资源消耗的性能测试也是又区别。个人觉得核心区别就这些了。

55、如何测试APP的流量和电量消耗情况？

- 1 在之前的公司里面我是使用的GT去测试APP使用过程中的流量和电量消耗。使用GT启动被测试的APP，流量基于NET指标来进行监控，一般会选择查询数据量或者页面请求量较大的页面去进行操作，一般操作的5~10钟左右，将GT收集到的数据通过adb提取到本地进行分析，通过的标准我们暂时是和同类型的app类似的功能操作的流量消耗数据进行对比，不超过即可，如不通过还会利用GT自带的TCPDUMP插件来搜集到请求的数据，查看传输数据信息。电量的消耗的话，利用GT中的I、U等指标参数来进行搜集，通过的标准和流量一样，和类似APP的进行对比。该块测试的次数较少。另外我还了解到adb指令也可以直接对这些内容进行测试。

56、APP兼容性测试测试了哪些型号？如果手机不够？你会如何处理？

- 1 我之前的公司团队比较小，而且对于APP的兼容性要求不是特别高，基于需求层面首先要求andorid和iso两个系统。版本没有特别强调。公司采购了部分二手机：OPPO、iphone8和华为Mate30，加上我们自己同事的手机小米等。除了这些真机以外，偶尔会使用一下模拟器和云手机来进行测试。

57、APP端权限如何测试？

- 1 基于之前测试经验，我们的APP会涉及到手机权限功能点有扫码、拍照、安装等，一般情况下会基于这些调用本机硬件的功能去单独进行测试，验证权限允许和不允许的情况的APP功能的使用情况。

58、APP无响应或者闪退你会如何分析问题？

- 1 APP的ANR和Crash产生的原因有很多，手机本身的资源问题，也有可能是代码本身的问题导致。通常我采用的方式有几个：Fiddler抓包、ADB查看日志和GT监控手机资源使用情况、查看服务器日志等方式。核心还是确认后台处理的接口和APP程序的代码是否有处理错误，日志可以查看一些错误日志信息：如Error级别的、EXCEPTION异常的。如获取到类似日志我自己会做一些简单的分析，1是确认功能接口信息，2是识别常见的异常：如数组越界和空指针等问题。如分析不出来，则会提交日志信息给开发进行确认，同时也会咨询开发具体问题，累积经验。

59、APP测试和微信小程序测试有什么区别？

- 1 其实APP和小程序本身基于功能业务的测试没有什么区别。非说业务上的区别的话，小程序的特性更偏向于小，所以在易用性的方面要求更高。第二：小程序的实现方式不一样，依托于微信终端，所以在兼容性的方面：还要考虑到各种微信版本得兼容。其它的我个人认为区别不大。

60、安卓系统和ios苹果手机系统测试有什么区别？

- 1 在以前项目中做兼容性测试的时候总结了一点小区别：在部分功能按键上有一些区别；分系统时iso端还需要多考虑下业务处理；UI页面跳转的形式不一样，iso界面对于空数据的处理等等

61、APP端接口测试和web端接口测试有什么区别？

- 1 个人认为是没有区别的，基于之前所做的项目来看。一个系统不同的客户端，我们对后台服务器是一个，也就是说后台服务代码是一致的，同样的功能对应同个接口。除非单边所特有的一些功能需要额外追加接口测试的内容外，没有什么区别。

62、如何测试的APP性能？

- 1 个人理解APP的性能测试的话分为两块，一块是对于APP在使用过程中，对于手机设备本身的资源消耗情况。第二是后台服务器的处理性能。那对于APP本身对于手机设备的性能测试我做两个方面的性能测试：一个手工操作APP，使用GT来搜集和分析资源使用数据情况，如手机CPU、内存、网络等等性能情况，第二使用monkey来做稳定性测试，验证APP是否会crash。对于后台服务器的性能测试，和常规的性能测试没有区别，我们公司使用jmeter还实现的（这块儿到性能里面再学）。

63、如何测试APP安装/卸载和升级？

- 1 首先对于app的安装卸载和升级测试，我们在版本迭代的过程中，并不是每次都专门去进行测试。因为有了版本迭代本身我们就需要到不同的手机上面去进行安装最新的安装包。如果安装不成功当场就能发现问题了。一般情况下比较大型的版本更新，我们会专门的去尽量的找全各种手机型号进行安装，同时还会考虑各种情况下的安装：如正在使用时、取消安装、安装中断、各种渠道安装、安装位置等，卸载的话除了正常卸载外，还会考虑安装文件、缓存文件清理的问题。升级会考虑各种渠道升级，跨版本升级等等

64、在之前项目中你是如何执行的交叉性测试？

- 1 这个测试过程中安装卸载的执行方式也差不多，不会每次版本都会进行测试。在之前的APP测试过程中的话，也是相对大点的版本才会专门安排交叉性测试的内容。核心的话测试：app在使用的过程中的其它一些影响性的操作，如通话、网络切换和横竖屏幕切换等。如果是播放性的业务还会考虑一下蓝牙设备、音量控制等等测试点

65、测试右移和测试左移分别用了哪些技术？

- 1 个人对于测试右移和测试左移的理解是这样，左移的话更多的是让测试工作提前，同时对于技术的要求和对开发服务架构的认知要更加深刻，如接口测试技术、单元测试技术等，同时基于质量体系中项目前面的阶段把控要更加深入：深刻理解需求，保障需求高质量，对于开发质量交付也需要更加严格，如开发自测、开发过程交付物的产出质量：接口文档、设计文档；这些都需是左移的内容。右移的话个人觉得可以是UI自动化技术、发布风险控制（灰度上线和测试）、版本差异化检查等等技术。无论是左移和右移都是为了项目质量来保障，基于个人的话现在对于***移投入的学习时间更多。

66、有没有做过数据脱敏处理？怎么处理的？

- 1 前提：根据实际的项目来，如果项目涉及到大量客户隐私数据，则有。特别的是银行、金融类项目
- 2

67、在你的项目中是如何实施接口测试？

- 1 说明：实施接口测试的方式有很多，对于不同的项目，不同团队工作模式都有不同，下面的答案仅仅作参考
- 2 在我之前的项目中，走的敏捷开发模式。版本不断的迭代过程中，有时候在开发整体提交SIT测试前，测试人员会有比较充裕的时间。在这种情况下，我们会尽量测试左移，提前介入测试，寻求接口文档，基于本次版本迭代的核心功能接口使用POSTMAN的工具进行接口测试。第二由于公司的项目的特性也会有一些历史功能版本，同时在项目任务比较空闲的期间，也会使用代码形式来对于单接口以及接口流程编写接口自动化测试脚本，使用的是：python+requests库+git+jenkins框架，一般的情况下原有的核心接口所实现的内容一般不会发生变化，这种接口脚本库可以用例对于后台接口的代码进行回归测试，对于项目质量也会多一重保障。

68、你怎么使用的POSTMAN实施的接口测试？

1. 之前说到一般是在开发整体提交SIT测试前，测试人员会有比较充裕的时间，会使用POSTMAN快速的对本次测试范围的核心接口进行测试
2. 使用的方式：我会依据接口文档，在POSTMAN中划分功能目录结构，方便区分接口功能模块。第二，依据接口文档在POSTMAN中创建好对应的Request请求用例，配置好请求地址、Request Header、Body，确认正常请求能跑通后。第三，优化接口测试点，基于多参数、少参数、错参数进行测试点配置，利用全局变量以及参数化方式对于各种情况的数据进行测试。第四，添加断言，基于响应体，通过结果提取器，提取必要的响应体结果数据进行判断。通常情况下多个接口之间数据有依赖的，还会使用关联的方式来完善接口测试用例。

69、你测试的过程中主要测试哪些类型的接口？有什么区别？

- 1 之前的项目核心还是以HTTP接口为主，也会涉及到部分HTTPS（当然这边实际工作中还有其它很多种类型的接口例如webservises，总之大同小异，会就说不会就不要硬搬）
- 2 HTTP和HTTPS核心区别，HTTP协议传输的数据都是未加密的，HTTPS是加密的，HTTPS协议是由SSL+HTTP协议构建的可进行加密传输、身份认证的网络协议，要比http协议安全。https协议需要证书。HTTP默认端口80，HTTPS默认端口443。

70、代码形式的接口测试你是怎么实现的，和Postman/jmeter实现接口测试有什么区别？你觉得哪种形式更好？

- 1 这里我有使用过python的第三方库requests库来进行的代码形式的接口测试。对于系统接口使用request所提供的get、post等方法来进行封装。并统一在一个Factory工厂类中进行实例化对象。然后通过单元测试框架pytest来组织测试用例。利用parameterized插件来实现参数化。以及使用pytest自带的assert来进行接口测试结果断言。最后基于整体测试用例代码使用git进行管理，结合jenkins来定时持续的运行。
- 2
- 3 Postman/jmeter基于做接口测试本身没有什么区别，因为本身接口实现形式没有发生改变，该配置的一样要进行配置。如果非要区别的话，1个Postman没有提供连接数据库的方式。而jmeter可以配置jdbc驱动来连接数据库。jmeter的工具更多的偏向于性能测试。如单单从只实现接口测试而言那Postman更加便捷易学，如还需要基于接口实现性能的话，那明显jmeter要强的多。

71、接口测试你核心校验的点有哪些？依据是什么？

- 1 第一个：站在系统或版本的层面我核心校验内容范围的主要是重点业务接口，以及流程
- 2 第二个：基于单接口而言，会验证正向和反向，反向会考虑到多参、少参、错参等情况，同时还会考虑请求头用户权限等
- 3 依据现有的接口文档，以及需求文档。接口应该返回什么数据其实要求都是根据需求文档的业务规则来的，和手工测试验证的结果没有区别，只是一个看的是界面，一个是看的json或者html的代码形式数据。

72、没有接口文档你怎么实施接口测试？

- 1 在我之前的公司，一般每次版本都会有接口文档，因为前后端需要进行联调。部分历史接口缺失会遗失掉接口文档，站在业务的角度我会先找出核心流程功能，使用F12的方式或者Fiddler抓包的方式获取到接口地址以及请求相关信息，如有部分字段具体含义不太清楚的话，则会找对应功能模块的开发人员进行确认，然后完善自己所记录接口相关信息。有了这些信息后，那后面接口测试实施和前面的描述没有区别。

73、第三方接口你怎么测试？

- 1 基于第三方接口的话，一般情况下会直接对接真实的第三方接口进行请求调用，获取实际响应数据后关联的到后续接口继续进行测试。部分无法直接调用的接口，例如支付，我们会使用python路由的方式来mock接口，模拟支付的各种返回情况。淡然也可以使用Postman来伪造接口返回数据，使用 Fiddler AutoResponder功能。

74、什么是重定向接口？加密的接口你怎么执行测试？

- 1 无非是请求的接口在接受到客户请求后，跳转到另外一个实际处理客户业务接口。有接触过部分加密的接口，我之前的公司对于部分敏感的接口就有进行过加密处理，加密的方式是我们团队自己的开发实现的一套加密方式。有专门的代码函数来进行处理，在测试这些接口时，我直接找开发拿对应的加密解密的函数对于请求信息进行加密后在调用接口请求，同时调用解密的函数来进行解密处理即可。

75、如何实现接口关联？说下实现方式？

- 1 在设计接口测试用例时，会设计一些业务场景性的测试用例。如果登录后需要进行下单；那么下单时需要用户登录的身份凭证，而用户身份凭证需要从登录接口返回的信息中进行获取。在jmeter中就需要用到数据提取器，我常用json提取器来获取响应信息，提取到结束信息后会形成一个变量。该变量则可以作为后续接口请求参数。一般情况下是这样实施的接口关联。

76、cookie、token和session有什么区别，简答说下看看？

- 1 session 是基于 cookie 实现的，session 存储在服务器端，sessionId 会被存储到客户端的cookie 中。
- 2 session 认证流程：
- 3 用户第一次请求服务器的时候，服务器根据用户提交的相关信息，创建对应的 Session
- 4 请求返回时将此 Session 的唯一标识信息 SessionID 返回给浏览器
- 5 浏览器接收到服务器返回的 SessionID 信息后，会将此信息存入到 Cookie 中，同时 Cookie 记录此 SessionID 属于哪个域名
- 6 当用户第二次访问服务器的时候，请求会自动判断此域名下是否存在 Cookie 信息，如果存在自动将 Cookie 信息也发送给服务端，服务端会从 Cookie 中获取 SessionID，再根据 SessionID 查找对应的 Session 信息，如果没有找到说明用户没有登录或者登录失效，如果找到 Session 证明用户已经登录可执行后面操作。
- 7
- 8 Token是服务端生成的一串字符串，以作客户端进行请求的一个令牌，当第一次登录后，服务器生成一个Token便将此Token返回给客户端，以后客户端只需带上这个Token前来请求数据即可，无需再次带上用户名和密码。最简单的token组成:uid(用户唯一的身份标识)、time(当前时间的时间戳)、sign(签名，由token的前几位+盐以哈希算法压缩成一定长的十六进制字符串，可以防止恶意第三方拼接token请求服务器)。

77、在你的项目中有多少接口？多少实现了接口测试？

- 1 这里需要有个基本认知，一般的情况下一个功能对应1个或多个接口。如购物车功能就会对应：加入购物车、购物车查询、购物车信息修改等接口，那具体一个系统中有多少个接口呢，也需要和自己简历中的项目功能的多少有直接关系。
- 2
- 3 例如B2C电商可以这样回答。在我之前的项目中的接口数量的具体数目不太清楚，在500-1000的样子(可以任取区间中的数量)。只对于核心功能模块的HTTP外抛接口进行了接口测试。实现了200多个，如各种情况的下订单、加入购物车、商品查询等

78、你是如何设计接口测试用例的？

- 1 在之前的项目中，对于接口测试用例设计层面包含两个方面：一个是基于单功能接口，一个是从业务场景出发进行设计。
- 2 基于单功能接口会考虑多参、少参、权限、错参等情况，如下单的接口：就会考虑各种前置，如使用优惠券下订单、使用积分抵扣下订单、基于库存商品下订单等等
- 3 基于业务功能场景：会站在用户角度考虑各种功能流程，如之前系统中实现的了从：普通会员登录-搜索商品-加入购物车-提交订单-mock支付-确认收货流程；普通会员-选择抢购商品-加入购物车-使用优惠券-提交订单流程等
- 4 最终确认后使用的jmeter or python+requests+unittest框架进行实现。

79、你执行接口测试的过程中有没有发现过BUG？简单举例说明一下？

- 1 有的。
- 2 例如在之前测试新增商品时，测试必选项少参的情况，期望返回对应msg提示信息，实际返回了NullPoint；
- 3 再例如基于提交订单，测试多张优惠券同时使用，错误参数，期望结果返回msg错误优惠券，实际返回了Outoff range

80、当一个接口出现异常时候，你是如何分析异常的？

- 1 个人理解对于外抛接口的测试场景其实和功能测试设计思路是一致的，无非是对于各种业务场景或各种异常数据录入角度去验证功能在处理的时候后台处理业务逻辑是否正确。所以当接口处理异常时，无非业务处理逻辑规则错误、或数据异常、或权限限制错误等。所以在分析的时候非站在需求的角度去分析接口响应数据是否正确。同时有时候也会接口请求日志去进行简要分析具体的错误原因。

81、有没有测试过webservice的接口？怎么测试的？

- 1 没测试过也没关系。可以直接跟面试官说明，对于webservice在实际测试过程和HTTP测试方式没啥区别，无非也是先确认好接口地址、请求信息等。通过postman测试的方式一致，对应配置方式没有任何区别。只是接口在实现层面有一些区别而已。在地址上面url一般就一个**wsdl，通过请求信息设定请求的方法和测试数据。

82、接口测试过程中如果需要依赖第三方java包，该如何实现接口测试？

- 1 在实际测试过程，除了直接通过jmeter配置接口请求外，偶尔也会出现需要使用到一些外部的代码来处理数据，例如调用外部的代码来对于数据进行加密解密。再例如部分非外抛的HTTP接口的测试。实现的方式了：获取到jar包后，直接将包放至 jmeter扩展包目录下，然后在测试计划中，在"add directary or jar to classpath"后面点击"添加"，选择对应jar包，打开，即可成功添加到jmeter中。根据需要可以使用beanshell来进行一些代码上的调用或处理。

83、Jmeter和Postman测试接口，你觉得哪个比较好？两者有什么区别？

- 1 我个人觉得各有的好处，相对于做接口测试来讲，Postman更简单易学容易上手，从实现接口测试来说两者并无太大差异。Postman本身就是专门用来做接口测试的，而jmeter更多的是基于接口来测试性能。

84、有没有做过性能测试？简单的描述一下当时是怎么测试的？

- 1 (基于操作层面级别回答)
- 2 1、有的，我在之前的项目测试过程中，对于项目核心的功能模块有按项目要求对于核心的功能模块实施过性能测试。如电商(商品搜索相关接口、购物车查询、提交订单等场景)，这里需要自己根据自己的项目来进行功能模块的挑选。
- 3 2、确认好后，会制定好性能测试场景，如单接口并发、压力测试场景。
- 4 3、然后部署好的性能测试环境，和生产环境等比资源搭建
- 5 4、使用jmeter工具，配置对应接口场景，设定好对应场景所用的定时器，使用nmon监控硬件资源，jmeter的聚合报告查看软件指标
- 6 5、最后总结性能结果产出报告即可

85、测试的什么场景？吞吐量多少，其它的指标结果如何？

- 1 在之前的项目测试过程中(这里拿电商说明)。主要测试了登录、商品搜索相关接口、购物车查询、提交订单等单个场景，以及混合操作场景。登录场景的话当时在测试环境所测试QPS在200左右，搜索商品大概是180左右。平均响应时间都在1s以下，具体的0.几秒大家可以适当脑补。

86、如何设计性能测试场景？

- 1 我会从几个层面来考虑性能测试场景的设计。第一个用户操作量大功能场景所涵盖的接口，第二个基于需求明确指出功能，第三个就是数据量较大接口。基于这些功能所涉及接口设定单接口并发场景，同时还会设计负载测试场景，最后会制定混合场景并行压力测试。

87、如何需求文档没有给出性能指标，该如何计算性能指标？

- 1 1、当时我在执行测试的时候需求层面没有特别全的指标要求，只提出了一个点，咱们的网站在客户网络情况OK的前提下，响应速度不能超过1s。拿到这个要求后我是这样做的。
- 2 2、基于系统运营数据获取到用户操作量最大的功能模块，但是选择登录、商品搜索、查看商品详情页、打开首页、购物车等，第二获取到了最大在线用户数，当时最大在线用户为5000，每个用户平均使用系统时间1小时，一天里面用户集中使用时间段为12个小时左右。基于该数据估算出最大并发用户数 ≈ 478 ($5000 * 1 / 12 \approx 417$ $417 + 3 * \sqrt{417}$)，基于计算最终定下来一个预期指标：500并发。实际测试环境会和生产环境的资源有偏差，最终等比取值100并发。
- 3 3、那到这个点实际我们已经计算出来了对应的单接口的指标数据了：VUSER 100 PER SERCODN<1 QPS:按照标准工时计算出来很低，所以取了一个基准值 20
- 4
- 5 (这里仅供参考)

88、压力测试、负载测试、并发测试有什么区别？

- 1 压力测试：压力测试的目标是测试在一定的负载下系统长时间运行的稳定性，尤其关注大业务量情况下长时间运行系统性能的变化（例如是否反应变慢、是否会内存泄漏导致系统逐渐崩溃、是否能恢复）；
- 2 负载测试：目标是测试在一定负载情况下系统性能（不关注稳定性，也就是说不关注长时间运行，只是得到不同负载下相关性能指标即可）；实际中我们常从比较小的负载开始，逐渐增加模拟用户的数量（增加负载），观察不同负载下应用程序响应时间、所耗资源，直到超时或关键资源耗尽，这就是所说的负载测试，它是测试系统的不同负载情况下的性能指标。
- 3 并发测试：主要指当测试多用户并发访问同一个应用、模块、数据时是否产生隐藏的并发问题，如内存泄漏、线程锁、资源争用问题，几乎所有的性能测试都会涉及并发测试。

89、除了使用jmeter收集性能测试结果数据还用过其它的方法吗？简单说下？

- 1 有的，之前的测试过程中还有用过nmon来搜集系统资源使用情况的数据，直接在服务器上面安装nmon的包，在启动测试的时候直接通过nmon命令来收集数据即可，会生成nmon监控数据文件，并获取到本地，通过nmon的宏工具来进行解析。当然除了这个搜集方法以外其实还有很多其它的数据监控方法：如jprofiler、dynatrace。

90、有没有搭建过性能测试环境？性能测试环境需要注意一些什么？

- 1 有搭建过的。例如之前的项目中就有搭建过一次，总共3台机器，2台应用服务器tomcat，1台数据库服务器mysql，负载均衡nginx的服务放在其中的一台应用服务器上。性能测试环境的话，个人觉得首先必须保障其专用性不能被其它因数干扰，第二尽量在局域网内，忽略的网络的影响，第三尽量和生产环境服务器架构一致（方便性能指标等比计算）。

91、你使用jmeter执行性能测试常用的定时器有哪些？有什么区别？

- 1 用的最多是：固定定时器和同步定时器。固定定时器，设定每个线程请求之前的等待时间，模拟多用户在实际操作的时候的等待时间。同步定时器，可以模拟并发设置一个阈值（请求数量），让多个用户能在同个时间点同时发出，模拟并发。

92、在性能测试过程中有没有发现过什么问题？举例说明？如何解决的？

- 1 (基于初中的问题进行说明)
- 2 有过的，在之前实施测试实施的过程中有遇到过内存泄露和数据库连接池配置的问题。内存泄露当时是这样的，在测试查询商品做并发的测试时候，几次短时间的并发测试在服务器硬件资源数据搜集的层面除非内存有一些增长外，其它指标数据都是满足条件。后续在做长时间持续负载测试的时候，发现当用户并发数稳定后，内存的时候持续性增加，处理完查询事务后没有任何的释放内存迹象，直到内存占满，最终导致服务器请求全部500。重启服务器后内存又恢复正常。后面该问题提交给开发进行定位确认，查询商品接口在处理完查询事务后，线程并未为释放，一直占用着资源所导致，具体的代码修复过程不说明。

93、除了jmeter做性能测试以外，还有没有其它的方式来实现？简单说下？

- 1 性能测试的方式有很多种，个人还有了解过LoadRunner、Locust等实现方式。LoadRunner的话相对jmeter而言支持更多形式的协议，例如socket、数据库协议ODBC、邮件SMTP协议等，同时还支持脚本录制。其功能强大没话说，但是需要破解，而且对于系统的兼容性不是太友好。Locust是一个python库，需要一定的编码功底。JVM本身也自带性能测试的功能。

94、TPS和QPS的区别是什么？

- 1 个人理解：TPS-Transaction per second 每秒处理事务数，从业务功能角度来讲，可以当做是一个功能接口合集，1个事务可以包含多个接口请求。如打开首页-会查询banner、查询楼层商品、会查询销售排行榜等，可以定义为一个事务，在Loadrunner工具中这个体现的很明显。当然从不同的角度事务的理解不一样，例如JVM，是从前台发起到查询数据到处理数据返回为一个事务。QPS:就question per second 每秒处理请求数，从翻译过来就可以看到区别了。

95、你们公司用户体量多大？最大在线有用户数有多少？

- 1 作为面试题来答的话，这里咱们不适合说过于庞大系统（你知道的！！）前面的题目84题提到了，可以说明下日活2w左右，最大并发也就5000。

96、在性能测试过程中如何准备足够量级的测试数据？

- 1 在之前测试的过程中我用过两种方式，例如在测试查询商品时，我们需要保障有足够量的商品数据。第一种方式是通过批量插入数据库的方式，参考27题。第二种就是直接通过jmeter调用新增商品的接口，通过参数化的方式批量生成测试数据（这种方式生成的数据更加完整）。

97、什么是数据库事务？事务有哪些特性？

- 1 可以理解为了完成一个业务动作所产生一系列数据库操作SQL的集合。如生成订单要查询库存、要操作订单表、要操作用户表等。一般一个数据库事务要保持原子性、一致性、隔离性、持久性（这里大家可以适当的去查一下资料进一步了解）

98、你们公司的服务器有没有做过集群？简单的说下？

- 1 有做过集群的，在之前的面试题还有介绍到性能测试环境会尽量模拟实际的生产环境的情况。在web服务器的层面会做应用集群，使用的是Nginx，通过配置分发规则来让部署在不同服务器硬件上的软件服务器来处理工作。我们所采用的分发机制是：最优资源。也有做数据库集群。

99、提高软件处理事务数最简单快捷的方式是什么？

- 1 一般事务的处理会涉及到数据的处理。个人任务两种方式吧，简单暴力的方式就是加服务器资源。第二个就是利用缓存机制来提高处理速度。

100、有没有提出过性能优化方案？

- 1 如果有能力才考虑。如果只是初中级测试工程师，能提出集群做分布式意见足够。

101、如何确定系统最大负载？

- 1 我在之前的性能测试实施过程中，是通过不断加压的方式来测试出系统的最大负载的。例如搜索商品在资源不变的情况下，不断提高并发用户数。例如10-30-50-100-150，直到测试到QPS、TPS、平均响应时间拐点。在资源满足的情况下出现拐点的话，既系统技术层面最大支持负载情况了。

102、有没有测试过百万级用户以上的性能测试场景？简单的描述下？

- 1 没有测试过，但是可以适当的表现出自己思路（量力而行）。百万级用户对信息有哪些，第一个用户量大，第二个数据量大。如果让我去测试的话，首先需要想清楚如何解决模拟庞大用户量的问题，第二数据如何构造的问题。那第一个问题，基于测试环境也是无法满足测试的要求，可以适当的等比降低服务器，那同样的并发用户数也会降低。Jmeter的常规情况下在我们普通的肉机上也就不超过200并发了，可以通过分布式的方式提高一下并发数量。当然也可以使用其它的方式如：Ignareo、IOCP等解决方案。大批量测试数据的准备，可以通过编写存储过程的方式，也可以直接导入生产环境数据来进行的方式。数据库的层面的话，通常都会有缓存机制。这些问题解决了之后，其实后续测试执行过程个人觉得大同小异。

103、你常用的Linux命令？

- 1 这里随便罗列一些：
vim tail -f ps -aux | grep 应用名称
2 top find / -name ** head df -hT
3 lsof mv cp 等

104、有没有用过数据库索引，有啥作用？

- 1 在之前的工作中没有实际的去具体创建过索引。但是我知道我们项目中所使用的数据库肯定是有加入了索引的。个人理解就是数据库管理系统中一个排序的数据结构，以协助快速查询、更新数据库中数据。如商品数据表，如果有了索引实际在查询的时候先查符合条件的索引数据，再去提取具体的商品数据。而不会整个商品表中筛选数据。

105、有没测试过后台，简单说下？

- 1 这个问题很多同学遇到过，首先做一个明确。后台分页系统前端业务后台，以及技术实践后台（实际也就是应用服务器层面）。说白了这里面面试官想问的无非是对于后台服务的测试，那涵盖的面就很广了，接口、性能、单元测试、中台测试、数据库、缓存、搜索引擎等等的测试都属于该范畴，而很多碰到这种问题就不知道如何回答。实际从中提取两个拿手或掌握的内容进行说明即可。

三、业务类

业务类需要学生自己更多对自己项目中的业务进行了解，不能把内容总是停留在登录/注册/购物车这些模块内

业务类-电商：

1、你负责的模块有哪些？请简单的描述下你是如何测试库存？

考虑点:

- 库存来源:客户退货（取消订单）、提前采购的库存
 - 取消订单、退款退货、修改库存
- 库存的消耗：购买、盘点损耗
 - 支付、盘点
- 库存锁定：
 - 下单、生成活动商品
- 库存商品价格

2、项目开通了哪些支付渠道？你是如何测试各种支付结果状态的？

- B2B一般不会打通线上支付,因为金额过大
- B2C和O2O一般说下微信/支付宝(2选1),个别银行即可
- 基于各种支付结果状态可以直接考虑mock，也可以真是对接实际支付环境来测试。

3、搜索引擎有没有测试过？简单的描述下是如何测试的？

4、你们的项目有哪些类型优惠券？测试哪些规则？

5、团购活动如何进行测试？参团、成团规则？价格是如何计算的？

6、有没有测试过埋点？如何测试的？

7、什么是SPU/SKU，阐述一下？

8、简单的描述下退款流程？退款失败怎么测试？

9、订单有哪些状态？你是怎么测试的？

10、推荐的业务规则是什么？

业务类-金融：

1、你负责的模块有哪些？简单的描述下整个系统的核心业务流程是什么？

2、借款人资格审核内容包含哪些？由谁来进行审核？

3、借款人的额度是根据什么来决定的？

4、借款的类型有哪些？分别是什么？区别是什么？

5、借款额度如何测试？

6、投资人资格审核内容有哪些？如何测试投资人充值功能？

7、在测试环境中如何测试放款是否成功？

8、如何测试还款功能？自动还款如何测试？

9、个人信用积分如何测试？

10、投资者收益如何计算？

四、笔试题

1、数据库-MYSQL

<https://www.cnblogs.com/donleo123/p/11642510.html>

2、手工笔试题

- 常用的测试方法有哪些？你最常用的方法是哪几个？分别举例？
- 请对于QQ聊天窗口设计一套测试用例？
- 请对于上传文件功能设计一套测试用例？
- 对微信朋友圈设计一套测试用例？
-

3、代码笔试题

- 计算出1~100的和

```
1 def sum_go(sum_to):
2     sum_all = 0
3     for i in range(1, sum_to+1):
4         sum_all += i * (-1) ** (1 + i)
5     return sum_all
6
7 if __name__ == '__main__':
8     result = sum_go(sum_to=100)
9     print(result)
```

- 定义一个空列表，给空列表添加10个数，使用循环判断将列表中的数据从小到大排列

```
1 list = [13, 22, 6, 99, 11]
2
3 for m in range(len(list)-1):
4     for n in range(m+1, len(list)):
5         if list[m] > list[n]:
6             temp = list[n]
7             list[n] = list[m]
8             list[m] = temp
9
10 print list
```

- 99乘法表

```
1 for i in range(1, 10):
2     for j in range(1, i+1):
3         print('{}x{}={} \t'.format(j, i, i*j), end='')
4     print()
```

- 其它

黑马程序员 <http://www.itheima.com>